

AMENDMENTS TO THE SPECIFICATION

Please amend paragraph 0007 as follows:

With the increase in Internet use has come an ~~the~~ increase in of online advertisements, online information dissemination, and user security concerns. Online advertisements are of a number of forms, including banner ads displayed across a user's screen or animated ads displayed in a special box or in a pop-up box on the screen. Usually, these types of ads are shown randomly or in a sequential order to all online users.

Please amend paragraph 0030 as follows:

R/O stands for required or optional. Each rule clause is either a required rule clause or an optional rule clause. Required rules operate like a Boolean AND function and optional rules operate as an OR function. For example, if a specific action has three required rules clauses associated with it, all three rules clauses must be true in order for the action to be taken. If a specific action has three required rules clauses and two optional rules clauses, all three required rules clauses, and at least one of the two optional rules clauses, must be true in order for the action to be taken.

Please amend paragraph 0033 as follows:

Grammar file ~~files~~ 304 defines ~~define~~ the comparison statements and associates the comparison statements with a specific [[,]] type. For example, grammar file 304 might define the comparison state \geq as an integer equal to or greater than and associate it with a demographical type, such as age. The following is an example of a ~~an exemplary~~ grammar file:

Type 1

integer equality

not equal to

starts with

Type 2

string equality

string not equal to

Please amend paragraph 0038 as follows:

Client rules engine 218 takes the rules ~~[[the-rules]]~~ clauses 302 and user data 306 and, using the grammar file 304, evaluates the rules by performing a rules based comparison of the user data to the values in a rules clause. The result is either a logical true or false result. As discussed previously herein, rules can be required rules or optional rules. A series of required rules behaves ~~behave~~ like a series of logically ANDed statements while optional rules act like ORed statements. Once a series of rules clauses 302 evaluates ~~evaluate~~ out correctly, an action occurs.

Please amend paragraph 0045 as follows:

FIG. 5 is a flowchart illustrating the operation of the rules engine according to one embodiment of the present invention. In step 502, demographical information is provided from client to server. This is the typical case where ~~were~~ filtering will be done at both the client and the remote server. In another embodiment, a remote server may send down packages without any filtering at the remote server. The remote server performs filtering at the server side using a remote rules engine. This determines which packages are to be sent to each client. For example, a remote rules engine may evaluate a rules clause ~~clauses~~ that compares the user's zip code to a specific zip code. If the zip codes match, a certain package is sent to the client computer.

Please amend paragraph 0055 as follows:

Fig. 7 illustrates how the embodiment of Fig. 6 processes data. The rules-based agent starts by loading Rule 1, shown as 710. Rule 1 begins with the clause Trigger = Trigger 1. The rules-based agent finds the piece of code corresponding to Trigger 1 in the system that meets this footprint, which is shown as 720. In this example, Trigger 1 is code that detects when a print job has been completed. Thus, when a user prints something, Trigger 1 sends a “fire” signal to the rules-based agent. The rules-based agent responds to this “fire” by proceeding to the second rules clause, which is an arithmetic evaluation to determine if the current ink level of the printer is lower than 10 percent of the maximum ink level. This clause contains two separate pieces of data that the rules-based agent needs from the data provider ~~[[,]] shown as 730~~. Thus, the rules-based agent goes to data provider 1, shown as 732, which will read the ink level of the printer and supply the printer present ink level data to the rules-based agent. The rules-based agent then goes to data provider 2, shown as 735, which will find that general information of the system and supply the printer maximum ink level data to the rules-based engine. These data are sent to the rules-based agent, which then performs the arithmetic function by dividing the present ink level by the maximum level and comparing that result to 0.1. If the result is less than 0.1 (i.e., the ink level is less than 10 percent of the maximum), the rules-based engine sends a signal to Action 1, shown as 740. Action 1 has code that may communicate with a display agent within the system to display to the user a message, such as “Your printer is low on ink, click here for more information.”